The Recommendation System Based on Semi-Supervised PSO Clustering Algorithm

ZHOU Wen Min[1,a],PAN Xiu Qin[1,b],LI Rui Xiang[1,c,*],Lu Yong[1,d]

[1] Information Engineering School of Minzu University of China, Beijing 100081, China

[a]zhouwenmin@wo.cn, [b] sandycode@163.com, [c] ruixiang0822@163.com,

[d] yl_cun@yeah.net

*Corresponding author

Keywords: Collaborative Filtering, Recommendation System, Particle Swarm Optimization, Semi-Supervised Clustering.

Abstract. Recommendation system has been one of the subject of intense in Computer Science, and it is widely used in Information Science. Various types of e-commerce systems and a large number of Internet applications need to use recommendation system to support their service. Recommendation system is aimed to provide users with the most valuable reference information. It filters out a large amount of useless information to help users to shorten the time to make a decision. A good recommendation system can accurately put the information to the specific users, and can accurately predict the users' behavior. In recent years, collaborative filtering recommendation algorithm has made a great progress. Among them, the performance of Semi-Supervised PSO clustering algorithm has been greatly improved to the traditional clustering methods. This paper tries to combine the Semi-Supervised PSO clustering algorithm with the clustering process of the recommendation system, and compare the performance of the new recommendation system with the old recommendation system based on the traditional clustering algorithm. The accuracy and effectiveness of the recommendation algorithm are validated by the experimental data.

Introduction

With the development of Internet application technology, information overload is becoming more and more serious[1].How to pick out the useful information for specific users and filter out other redundant information from the mass of information, becomes a hot and difficult topic of Internet technology studies.

Recommendation system is aimed to help users to select the valuable information for them, and filter out a lot of useless information, so as to realize the directional push of information.

Recommendation system is a kind of information filtering system, which can give advice to product information or services of interest to the users[2].When there are many alternatives available to users, the recommendation system will help users to select the most suitable products for them. Recommendation algorithm is the core module of the recommendation system, which is used to complete the main work flow of the recommendation system. At present, the mainstream of the recommendation algorithm are collaborative filtering recommendation algorithm[3], content-based recommendation algorithm[4],the recommendation algorithm based on the graph structure[5] and hybrid recommendation algorithm, etc. To a certain extent, the recommendation algorithm promotes the development of the recommendation system module of E-commerce websites, but it also exists classification accuracy error. The recommendation algorithm based on traditional K-Means clustering method [6] need specifying an initial value K as a parameter to select K User-Item vectors randomly. Then these K vectors are used as the initial cluster centroid to carry on the next step of the clustering process. The process of selecting User-Item vectors is easy to produce a certain clustering error. In the case that the Euclidean distance between each cluster centroid is large enough at the beginning, the error value may be relatively small.

On the contrary, in the case that the Euclidean distance between each cluster centroid is very small, the error value may be relatively large and clustering results may be bad. At the present stage, there are many researches reduce the error caused by the initial value dependence of the K-Means algorithm. To a certain extent, this improves the accuracy of the clustering process for the recommendation system. However, other reliable clustering algorithms can be used to further improve the accuracy of clustering process. This research uses Semi-Supervised PSO clustering algorithm for classification, then apply the clustering results to the recommendation algorithm. Finally, generate relatively reliable recommendation results. Simulation results shows our new approach outperforms the recommendation algorithm based on K-Means.

The second part of this paper introduces the relevant research foundation of the algorithm proposed in this paper. The third part will focus on the basic idea of the recommendation algorithm based on Semi-Supervised PSO clustering algorithm proposed in this paper and its implementation. The fourth part is the corresponding experiment and analysis of the algorithm. The last part is the conclusion and prospect of this paper.

Related Research

Clustering

It is to determine which entity belongs to class by calculating the similarity between entities. The principle for clustering is that the similarity among the entities within a class is larger, and the similarity among the entities is smaller. Generally speaking, Clustering is formally described as

$U=\{p_1,p_2,\ldots p_\mathrm{i}\}$

Where U represents entity set belong to n-dimension space. pi represents ith pattern, and is the set defined as $i=\{1,2,\ldots,n\}$,The clustering result is defined as the Eq. 1 and Eq. 2[11]:

$$\mathbf{U}_{t=1}^{k} C_t = U \quad (\mathrm{t}=1,2,\ldots,\mathrm{k},)$$

$$To \forall C_m, C_r \subseteq U, C_m \neq C_r, Then C_m \cap C_r = \varnothing$$

$$\text{(Only for Rigid Clustering);} \tag{1}$$

$$MIN_{\forall P_{mu}\in C_m, \forall P_{rv}\in C_r, \forall C_m, C_r \in U \& C_m \neq C_r}(proximity(p_{mu}, p_{rv}))$$

$$> MAX_{\forall P_{mx}, P_{my}\in C_m, \forall C_m \in U}(proximity(p_{mx}, p_{my})) \tag{2}$$

Where C_t is the clustering result. The $proximity(p_{ms},p_{ir})$ is used to describe the similarity distance of patterns. The first index of function proximity represents the class which the pattern is part of and the second index of that is pattern in a particular class.

The key steps of clustering are listed as the following:

Step 1: feature extraction: Determine the effective features, then exact features.

Step 2: Compute similarity: Select appropriate distance function to calculate the similarity.

Step 3: Determine clustering criteria: Determine appropriate clustering criteria, then cluster or group data.

Step 4: Clustering result evaluation: Evaluate the validity of clustering results.

In this paper, the Euclidean distance is used to calculate the similarity. Then the Euclidean distance between x and xi can be computed through the Eq. 3.

$$dis = \|x - x_i\| = \sqrt{\sum_j (x_j - x_{ij})^2}, \text{ where } x_{ij} = |x_j - x_i| \tag{3}$$

Where $\{(x_i, y_i)\}$ is feature vector, as well as x_i and y_i is the d-dimension vector, respectively.

Particle Swarm Optimization

In PSO, each particle in the swarm represents a potential solution. For a particle, it has two vectors, velocity and position. Particles fly in the search space with n-dimension according to its personal best position (*pbest*) and the global best position (*gbest*) found by all particles. The movements of particles are defined by the following two equations , Eq. 4 and Eq. 5 (Kenndy and Eberhart, 1995).

$$V_i(t+1) = w \cdot V_i(t) + c_1 \cdot r_1 \left(pbest_i - X_i(t) \right) + c_2 \cdot r_2 \cdot \left(gbest - X_i(t) \right) \tag{4}$$

$$X_i(t+1) = X_i(t) + V_i(t+1) \tag{5}$$

Where Vi and Xi are the velocity and position of the *i*th particle separately. ***pbest**$_i$* and ***gbest*** are the personal best particle of the *i*th particle and the global best particle, respectively. Set ***pbesti***=(x_{i1},x_{i2}….x_{in}), and ***gbest***=(g_1,g_2,…,g_n). The parameter ω represents an inertia factor, and r_1 and r_2 are two random numbers independently generated in the range of [0,1], and c_1 and c_2 are two learning factors used to control the influence of the social and cognitive components. In order to maintain PSO stability, a velocity threshold $V_{\max}$ is used to limit the value of velocity.

The main steps of PSO are described as follows:

Step 1: Initialize the position and velocity values of the particle and swarm, and give the initial value of the parameters such as the ω, *c1, c2, r1, r2* and so on.

Step 2: Calculate fitness values of all particles in the swarm according to the result of Step 1 and the fitness formula.

Step 3: Initialize and calculate the personal best value of each particle and the global best value of the whole swarm for the first time.

Step 4: Execute the loop (Condition: The number of iterations did not reach the specified number of times.)

{

Sub-step 1: Update the velocity and position of each particle.

Sub-step 2: Use fitness function to calculate the fitness value of each particle in the swarm.

Sub-step 3: Update the personal best value of each particle and the global best value of the whole swarm according to the computing result of the fitness value of each particle in Sub-step 2.

}

Semi-Supervised Particle Swarm Optimization

The main idea of Semi-Supervised learning is to add unlabeled samples to labeled samples to improve the clustering performance. How to make comprehensive use of labeled samples and unlabeled samples is the main problem to be solved in Semi-Supervised learning.

The formal description of Semi-Supervised learning is [7][11]:A sample set *S*=*UL* from an unknown distribution, where *L* is a labeled sample set, *L* ={(x_1,y_1),(x_2,y_2),…,(x_L,y_L)}, *U* is an unlabeled sample set, *U* ={x'_1,x'_2,…,x'_U}, and we hope to find a function *F* which satisfies the conditions described as follow:

F : *X*->*Y* can accurately predict the label *y* for the sample *x*.

Where x_i, x_j□*X* are *d*-dimensional vectors, y_t□*Y* is the label of the sample x_i.

Combined Semi-Supervised learning clustering method with swarm intelligence, the computational method of its fitness value is shown as Eq. 6 [8] :

$$Fitness(p_{id}) = b \cdot (\tfrac{1}{l} \sum_{j=1}^{l} d(x_j, p_{id,CL[j]})) + (1-b) \cdot [\frac{1}{u} \sum_{k=1}^{u} \min\{d(x_k, p_{id,0}), d(x_k, p_{id,1}), \ldots\ldots d(x_k, p_{id,CL-1})\}] \tag{6}$$

Where *CL*[*j*] represents the value of label corresponding to the vector x_j. $p_{id,CL[j]}$ represents the centroid vector of *id*th particle who has the same label as the label of x_j. $p_{id,CL\text{-}1}$ represents the centroid vector of *id*th particle computed at current time whose label equals to *CL*-1. *d*(x_j, $p_{id,CL[j]}$) represents the Euclidean distance between labeled sample vector x_j and the centroid vector in *id*th particle whose label is the same as vector x_j. The *d*(x_k, $p_{id,CL\text{-}1}$) represents the Euclidean distance between unlabeled sample vector x_k and the centroid vector of *id*th particle whose label value equals to *CL*-1. *l* represents the total number of labeled samples. *u* represents the total number of unlabeled samples.

Semi-Supervised PSO algorithm generates *n* particles from the start. Each particle carries *CL* *d*-dimensional vectors, and represents the centroid vectors corresponding to these *CL* labels. By iteration, it is necessary to find the optimized position corresponding to their label respectively. After learning iterations, the path owned by these *n* particles represents the position of centroid vectors corresponding to each label. They gradually converge to the best position.

The Recommendation System Based on Semi-Supervised PSO

Using Semi-Supervised PSO Clustering Algorithm for Classification

First, we should score the item ratings of different users in User-Item vector, a two-dimensional array, each item of User-Item[*i*][*j*] represents a specific user I rating for a particular item J. Initial value may exists a large number of 0 points, which means that the user hasn't bought this product or rated this item. Then we need to cluster these User-Item vectors, so that the users have higher similarity are in the same cluster and the users have lower similarity are in the different clusters. The centroid of two different clusters maintain some distance. We use Semi-Supervised PSO clustering algorithm to cluster the User-Item vectors, and SSPSO need to use a small number of labeled samples and a large number of unlabeled samples. Therefore, in order to facilitate the classification of SSPSO, the input User-Item vectors must satisfy a small number of vectors are labeled, and a large number of vectors are not labeled.

In the clustering process of SSPSO, we use the Eq. 6 to execute the clustering iteration.The value of Fitness is correlated with the proportion of labeled samples and unlabeled samples.In this formula, a parameter is represented as β and its value is in the range of [0,1].The algorithm is equivalent to the standard PSO clustering algorithm when the value of β is equal to 1. On the other hand, the algorithm is equivalent to the unsupervised PSO clustering algorithm when the value of β is equal to 0. A lot of experiments show that while the value of β is different, and the clustering results are often different, too. The selection of β has directly impact on the accuracy of the clustering results. Therefore, in order to achieve the clustering performance of self-adaptive parameter optimization, we introduce a new group of formulas shown as Eq. 7 and Eq. 8 to dynamically adjust the value of the parameter β.

$$Fitness(b_{id}^{n}) = (b_{id}^{n-1} + v_{id}^{n-1}) \cdot (\tfrac{1}{l}\sum_{j=1}^{l} d(x_j, p_{id,CL[j]})) + (1 - b_{id}^{n-1} - v_{id}^{n-1}) \cdot [\frac{1}{u}\sum_{k=1}^{u} \min\{d(x_k, p_{id,0}), d(x_k, p_{id,1}), \ldots\ldots d(x_k, p_{id,CL-1})\}], (n \geq 1) \tag{7}$$

$$v_{id}^{n} = wv_{id}^{n-1} + c_1 \cdot rand_1 \cdot (pbest_{id}^{n-1} - b_{id}^{n-1}) + c_2 \cdot rand_2 \cdot (gbest^{n-1} - b_{id}^{n-1}), (n \geq 1) \tag{8}$$

Sparsity Measurement and Recommendation Generation

After finishing the clustering process, we make further sparsity measurement according to the clustering results.Through the final prediction is a linear combination which calculated by the Eq. 9,we need to calculate the most suitable α, which represents the proportion of local neighborhood and global neighborhood in the prediction.Anand and Bharadwaj[9] introduced several methods for the measurement of sparsity, which calculate the proportion of the global neighborhood according to the active users and the ratings for the items rated by these users which need predicting. We use $\alpha1$, $\alpha2$, $\alpha3$, $\alpha4$ and $\alpha5$ to represent the results of five kinds of sparsity measurement methods. In which,$\alpha1$ calculates the sparsity level of the entire matrix, and this computational method is fixed for all users, shown as Eq. 10. $\alpha2$ uses the proportion of the users who rated the item to all users to calculate the sparsity level, which formula is shown as Eq. 11.$\alpha3$ uses the ratio of local neighbors and global neighbors to measure the sparsity level,which formula is shown as Eq. 12. Similarly,$\alpha4$ and $\alpha5$ could also calculate the sparsity level in a similar way, which formula are shown as Eq. 13 and Eq. 14.Through these five formulas above have both advantages and disadvantages in calculating the degree of the sparsity level, then $\alpha6$ and $\alpha7$ are proposed,which consider to take weighted average of these formulas.$\alpha6$ and $\alpha7$ are also called unified sparsity measure (UMS), which consider all sparsity measure formulas, shown as Eq. 15 and Eq. 16.

$$Pred(i,k) = a \times Pred_G(i,k) + (1-\alpha) \times Pred_L(i,k) \tag{9}$$

$$a_1 = 1 - \left(\frac{n_R}{n_U \times n_I}\right) \tag{10}$$

$$a_2 = 1 - \left(\frac{n_u}{\max\limits_{u \in U}(n_u)}\right) \quad (11) \qquad a_3 = 1 - \left(\frac{\mathrm{L_n}(u,i)}{\mathrm{G_n}(u,i)}\right) \quad (12)$$

$$a_4 = 1 - \left|\frac{\mathrm{L_n}(u,i)}{\mathrm{N_i}}\right| \quad (13) \qquad a_5 = 1 - \left|\frac{\mathrm{L_u}(u,i)}{\mathrm{L_u}}\right| \quad (14)$$

$$a_6 = \sum_{i=1}^{5} w_i \times a_i \text{ (Where , } \sum_{i=1}^{5} w_i = 1 \text{) } (15) \qquad a_7 = \sum_{i=1}^{5} a_i \left(\sum_{j=1}^{i} w_j \times a_j\right) (16)$$

In Eq. 10, n_R represents the total number of ratings, n_U represents the total number of exist users in the matrix User-Item, n_I represents the total number of items. In Eq. 11, n_u represents the total number of items which are rated by user u. In Eq. 12, Eq. 13 and Eq. 14, $\mathrm{L_N}(u,i)$ represents the local neighborhood set which satisfied ith item is rated by user u, $\mathrm{G_N}(u,i)$ represents the global neighborhood set which satisfied ith item is rated by user u. N_i represents users set that rated for ith item. L_u represents the local neighbors set of user u. Eq. 15 must satisfy the condition $\sum_{i=1}^{5} w_i = 1$.In Eq. 16, a_i is a constant and suitable for all values of variable i ,and $\sum_{i=1}^{5} a_i = 1$.

According to the prediction formula proposed by Resnick, we could predict the rating of ith item for an active user u[10],including the local prediction Eq. 17 and the global prediction Eq. 18.We can achieve the final prediction result by taking these two computation results into the linear combination Eq. 9.

$$\mathrm{Pred}_L(i,k) = \bar{r}_i + \frac{\sum_{j \in N_L(i)} L_{sim}(i,j) \times (r_{jk} - \bar{r}_j)}{\sum_{j \in N_L(i)} L_{sim}(i,j)} \tag{17}$$

$$\mathrm{Pred}_G(i,k) = \bar{r}_i + \frac{\sum_{j \in N_G(i)} G_{sim}(i,j) \times (r_{jk} - \bar{r}_j)}{\sum_{j \in N_G(i)} G_{sim}(i,j)} \tag{18}$$

In Eq. 17, $L_{sim}(i, j)$ is the local similarity value computed for user i and user j, $N_L(i)$ is the neighborhood set of user under consideration. $\bar{r}_j$ is the mean rating of user j. And r_{jk} is the rating of user j on item k. Similarly, in Eq. 18, $G_{sim}(i, j)$ is the global similarity value computed for user i and user j, $N_G(i)$ is the neighborhood set of user under consideration.

Next, we use unsupervised PSO algorithm to study and update the value of $\alpha6$ and $\alpha7$[10],so that the value of α has a best weight in the linear prediction Eq. 9.First, we randomly take values for ωi and ai in the Eq. 15 and Eq. 16,and generate N particles. Then use PSO algorithm to update the weight value of every part. The evaluation criteria is given by Eq. 19.Every particle generates a new fitness value by evaluation function. After the evaluation process, PSO algorithm will execute the iteration process until up to the stop criteria $\left|fit(t)-fit(t+1)\right|>e$ [10].

$$fit(p)=\frac{1}{T}\sum_{i\in T}\left|r_{ui}-\text{Pred}(u,i)\right| \tag{19}$$

Implementation Steps of The Recommendation Algorithm This Paper Proposed

Step 1:Load labeled sample data set and unlabeled sample data set into two independent arrays, respectively.

Step 2:Initialize the swarm with *n* particles carried β value. Use random numbers in the range of [0 1] as their initial values.

Step 3:Use β value of these *n* particles to execute SSPSO algorithm separately, and calculate the personal best value of each particle and the global best value of the whole swarm for the first time.

Step 4:Execute the loop(Condition: The number of iterations did not reach the specified number of times.)

{

Sub-Step 1: Update the velocity and the β value produced by the position of each particle.

Sub-Step 2: Use the new β value to update the formula of *SSPSO*.

Sub-Step 3: Load labeled sample vector data sets and unlabeled sample vector data sets and initialize them, and make sure that the range of values of each attribute meets the requirements of the algorithm.

Sub-Step 4: Initialize the swarm with *t* particles for SSPSO clustering algorithm. Use random number generation algorithm to generate random numbers which meet the requirements of the scope of the algorithm as the initial value of each particle. The data structure of the independent variables in each particle is a two-dimensional array, with *d* rows, *CL* columns.*CL* represents total numbers of labels, *d* represents the total number of attributes.

Sub-Step 5: Execute SSPSO algorithm to learn and gradually optimize the best position of these *C*L centroid vectors.

Sub-Step 6: Using the 1-NN method to assign labels to all unlabeled samples. The method is to make comparison between the sample vectors to be labeled and centroid vector of each label computed by SSPSO algorithm in Sub-Step 5 one by one. Then find its minimum Euclidean distance and assign the label of centroid vector found by the minimum Euclidean distance to the corresponding unlabeled sample vector.

Sub-Step 7: Calculate the fitness value of every particle in the swarm.

Sub-Step 8: Update the personal best value of each particle and the global best value of the whole swarm according to the computing result of the fitness value of each particle in Sub-step 7.

}

Step 5: Save the Final β value and clustering results,and compute the local similarity and global similarity values of all users.

Step 6: Update ωi and ai in Eq. 15 and Eq. 16 using unsupervised PSO algorithm.

Step 7: Take the optimal ωi and ai into the linear prediction Eq. 9.

Step 8: Output the recommendation results.

Experimental Analysis

A comparable experiment was carried out to test the effectiveness of the recommendation system based on Semi-Supervised PSO clustering algorithm and traditional K-Means clustering algorithm. Then verify the validity of the proposed algorithm. In the process of measuring recommendation results. We use the mean absolute error (MAE) and prediction accuracy to compare the performance of the two algorithms.

Selection of Experimental Parameters

In general, the parameters involved in the calculation of PSO are selected as follows: $c_1=2, c_2=2, r_1$ and r_2 take two independent random floating points range in [0,1].The value of the parameter ω can be represented by the Eq. 20.

$$w = 1 - \frac{curTimes}{T} \quad (20)$$

Where *curTimes* represents the number of iterations PSO algorithm has been executed, and *T* represents the total times of iterations.

Selection of Experimental Data

This paper uses the rating data of 3883 movies rated by the first 500 users in the MovieLens dataset as the experimental data. Each movie contains a name field and some types field, and belongs to at least one type. User data includes basic information such as gender, age, and occupation. Rating data contains the scores of specific films rated by some users. The ratings scale is in the range 1-5 with 1-“bad”to 5-“excellent”. The sparsity degree of the dataset is 99.2%.

Experimental Results and Analysis

This paper uses MAE and prediction accuracy to measure the performance of the recommendation system. To a specific user u_i ,MAE is calculated by Eq. 21.Where $|S_i|$ is the cardinality of the test ratings set of user u_i .The total MAE over all the active users, it can be computed as Eq. 22,where T represents the total number of users.

$$\mathrm{MAE}(u_i) = \frac{1}{|S_i|}\left|\mathrm{Pred}(i,k) - r_{ik}\right| \quad (21)$$

$$\mathrm{MAE} = \frac{1}{T}\sum_{i=1}^{T}\mathrm{MAE}(u_i) \quad (22)$$

MAE is the measure of the deviation of recommendations from their true user-specified values. MAE is computed by first summing up the absolute errors of the *n* corresponding rating-prediction pairs and then computing the average. The lower the MAE, the more accurate the method is. The advantages of this measurement are that its computation is simple and comprehensible and it has well studied statistical properties for testing the significance of the difference.[10]

We set the *k* value to 5 or 10 in K-Means. Similarly, we also set the number of classlabels to 5 or 10 in Semi-Supervised PSO algorithm. The values of MAE are derived through two different recommendation algorithms are shown in Fig.1 and Fig.2.In the graph, we can clearly see that the MAE value of the recommendation system based on SSPSO is significantly smaller than the recommendation system based on the K-Means clustering algorithm.

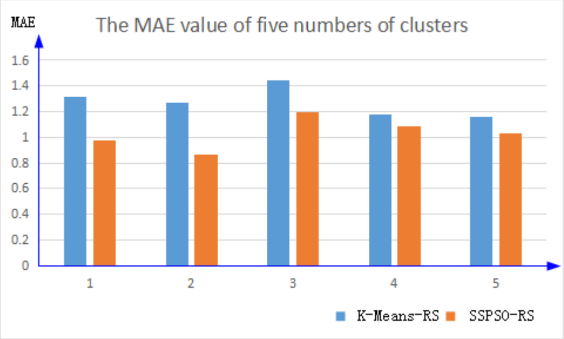

Fig.1 The MAE value of 5 clusters

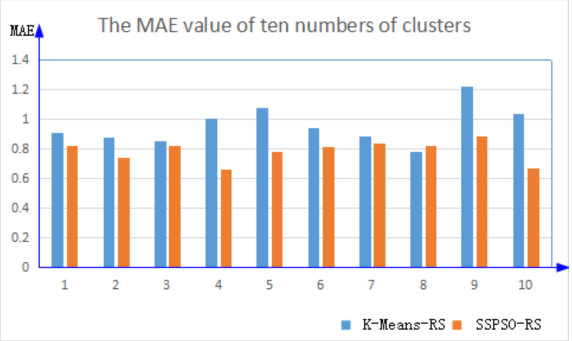

Fig.2 The MAE value of 10 clusters

The prediction accuracy is computed by Eq.23.

$$Precision = \frac{\sum_{u \in U}|R(u) \cap T(u)|}{\sum_{u \in U}|T(u)|} \quad (23)$$

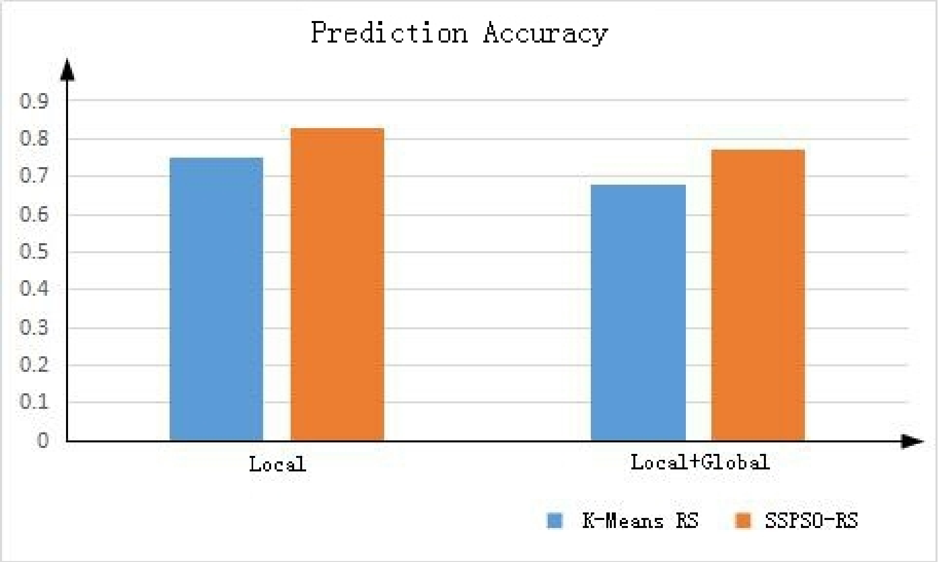

Fig.3 Prediction Accuracy

The prediction accuracy comparison chart of the recommendation system based on K-Means and based on SSPSO is shown in Fig.3.When the global similarity is under considered, the prediction accuracy is lower than the case that only local similarity is under considered. From the prediction results, the prediction accuracy of the recommendation system based on SSPSO is higher than the recommendation system based on K-Means.

In addition, the time complexity of the recommendation system based on SSPSO is relatively high. Therefore, the recommendation system based on K-Means have priority in time-consuming because of the shorter clustering iteration process time.

Conclusion

In this paper, an improved SSPSO algorithm is proposed and introduced into the process of recommendation algorithm. The new recommendation algorithm focuses on improve the prediction accuracy. Simulation results show the effectiveness and feasibility of the algorithm to improve the prediction accuracy from another aspect. However, the online optimization calculation process of the parameter in SSPSO costs a certain computing time and computing resources. So the algorithm costs more resources, especially for massive data samples. Further reducing the time complexity and space complexity is the improvement direction of the research in the future.

References

[1]Eppler M J, Mengis J. The Concept of Information Overload - A Review of Literature from Organization Science, Accounting, Marketing, MIS, and Related Disciplines (2004)[J]. Eres, 2008, 38(1):325-344.

[2] Qiuyong Yang. Research on the recommendation system of learning the best score by using the tensor decomposition method[D]. South China University of Technology, 2012.In Chinese [3]Breese J S, Heckerman D, Kadie C. Empirical Analysis of Predictive Algorithms for Collaborative Filtering[J]. New Page, 1998.

[4]Sarwar B, Karypis G, Konstan J, Riedl J. Item-based collaborative filtering recommendation algorithms[C]// International Conference on World Wide Web. ACM, 2001:285-295.

[5]Xi Wang, Shuangyan Duan. An improved recommendation algorithm based on the network structure of two parts[J]. Computer Application Research, 2013, 30(3):771-774.In Chinese [6]Macqueen J. Some Methods for Classification and Analysis of MultiVariate Observations[C]// Proc. of, Berkeley Symposium on Mathematical Statistics and Probability. 1967:281-297.

[7]Zhu X.“Semi-supervised learning literature survey,” Computer Sciences Technical Report 1530, University of WisconsinMadison, 2008.

[8]Zhang X, Jiao L, Paul A, Yuan Y, Wei Z, Song Q. Semisupervised Particle Swarm Optimization for Classification[J]. Mathematical Problems in Engineering, 2014, 2014(2):1-11.

[9]Anand D, Bharadwaj K K. Utilizing various sparsity measures for enhancing accuracy of collaborative recommender systems based on local and global similarities[J]. Expert Systems with Applications An International Journal, 2011, 38(5):5101-5109.

[10]Bakshi S, Jagadev A K, Dehuri S, Wang G N. Enhancing scalability and accuracy of recommendation systems using unsupervised learning and particle swarm optimization[J]. Applied Soft Computing, 2014, 15(2):21-29.

[11]Basu S, Banerjee A, Mooney R J. Semi-supervised Clustering by Seeding[C]// Nineteenth International Conference on Machine Learning. Morgan Kaufmann Publishers Inc. 2002:27-34.